Computationally Efficient Canonical Molecular Dynamics Simulations by Using a Multiple Time-Step Integrator Algorithm Combined with the Particle Mesh Ewald Method and with the Fast Multipole Method

MASAAKI KAWATA, MASUHIRO MIKAMI

Department of Physical Chemistry, National Institute of Materials and Chemical Research, Tsukuba 305-8565, Japan

Received 2 February 1999; accepted 24 September 1999

ABSTRACT: An efficient implementation of the canonical molecular dynamics simulation using the reversible reference system propagator algorithm (r-RESPA) combined with the particle mesh Ewald method (PMEM) and with the macroscopic expansion of the fast multipole method (MEFMM) was examined. The performance of the calculations was evaluated for systems with 3000, 9999, 30,000, 60,000, and 99,840 particles. For a given accuracy, the optimal conditions for minimizing the CPU time for the implementation of the Ewald method, the PMEM, and the MEFMM were first analyzed. Using the optimal conditions, we evaluated the performance and the reliability of the integrated methods. For all the systems examined, the r-RESPA with the PMEM was about twice as fast as the r-RESPA with the MEFMM. The difference arose from the difference in the numerical complexities of the fast Fourier transform in the PMEM and from the transformation of the multipole moments into the coefficients of the local field expansion in the MEFMM. Compared with conventional methods, such as the velocity-verlet algorithm with the Ewald method, significant speedups were obtained by the integrated methods; the speedup of the calculation was a function of system size, and was a factor of 100 for a system with 3000 particles and increased to a factor of 700 for a system with 99,840 particles. These integrated calculations are, therefore, promising for realizing large-scale

Correspondence to: M. Mikami; e-mai: mikami@home.nimc.go.jp

Contract/grant sponsor: The Domestic Research Fellowship (Japan Science and Technology Corporation)

CCC 0192-8651 / 00 / 030201-17

molecular dynamics simulations for complex systems. J Comput Chem 21: 201–217, 2000

Keywords: molecular dynamics; Nosé–Hoover; RESPA; particle mesh Ewald method; fast multipole method; macroscopic expansion

Introduction

Molecular dynamics (MD) simulations have been recognized as a powerful tool for investigating phenomena occurring in systems such as biological macromolecules, organic crystals, and complex liquids.[1] Because the calculation of the Coulomb interaction is the most time consuming part in MD calculations, it directly affects the efficiency of the simulation. Therefore, the size of the system that can be treated by MD simulations depends on the performance of the method used to calculate the Coulomb interaction. The method also seriously affects the accuracy of the MD simulation because the magnitude of the Coulomb interaction is stronger than any of the other interactions in the system (e.g., the van der Waals interactions). The "cutoff" techniques introduced to reduce the computational cost caused by slow convergence of the Coulomb interaction have been shown to produce unrealistic behavior in the simulation.[2] To prevent unrealistic behavior, the Ewald method has been widely used for treating the Coulomb interaction in periodic systems. Although the efficiency of the Ewald method can be improved by optimal choice of the parameters,[3,4] the cost is prohibitive for large systems because for a given number of particles in the system, N, the computational cost scales as N^2 for the conventional algorithm and as $N^{3/2}$ for highly optimized algorithms.[5,6] Therefore, more efficient techniques are needed to permit application of MD simulations to larger and more realistic complicated systems.

Recently, two techniques for accelerating the calculation of the Coulomb interactions have been proposed and successfully used to reduce the computational cost of MD simulations—the particle mesh Ewald method, and the fast multipole method. The particle mesh Ewald method (PMEM) is the most promising algorithm of the methods derived from the standard Ewald method.[3,6–10] Similar to that in the Ewald method, the total Coulomb interaction in the PMEM is evaluated as a sum of the contributions from the real space sum, the reciprocal space sum and the self-correction term. Whereas the real space sum and the self-correction term in the PMEM are calculated in the same way as in the Ewald method, the reciprocal space sum is evaluated using the fast Fourier transform techniques (FFT). The efficiency of the PMEM comes from the rapid evaluation of the reciprocal space sum and the computational effort of the algorithm scales as $N \log N$. An alternative technique to reduce the computational cost of the Coulomb interaction is the fast multipole method (FMM), originally proposed by Greengard and Rokhlin.[11–19] In the FMM, the simulation system is divided into progressively smaller boxes. The Coulomb interaction is evaluated as a sum of the direct interaction of each particle with other particles in the same smallest box and with particles in their neighboring boxes, and of the interaction with the local field, which is produced by the multipole expansion of the charge distribution of the distant particles. In the construction of the local field due to remote charges, the multipole expansion of the charge distribution and the hierarchical structure of the boxes are effectively used, which results in computational effort of order N. For consistent treatment of the Coulomb interaction in periodic systems, similar to the Ewald method, the FMM is required to include the effect of the periodic images of the system. The periodic FMM (PFMM) using the Ewald sum techniques has been developed by several research groups.[15,19] A different approach to include the periodic images is the macroscopic expansion of the FMM (MEFMM), in which the same techniques used in the evaluation of the local field in the original system are used to macroscopically assemble replicas located around the original system.[20] In the calculation with the PFMM, the effect from the periodic images is factorized and becomes independent of the particle coordinates only when the size or the shape of the system does not change. Thus, for simulations with such conditions, the PFMM is a promising method to reduce the computational cost. On the other hand, the MEFMM can be easily applied to systems that have anisotropic periodic boundary conditions, to which the Ewald technique is difficult to apply.

Another approach for accelerating the MD calculations is the multiple time-step integrator method

that is designed to reduce the frequency of the time-intensive parts of the simulation, such as the evaluation of the Coulomb interaction. In the multiple time-step integrator method, motion with a small period is evolved by using a small time step, whereas motion with a large period is evolved by using a large time step. This means that the calculations of the bonding interaction with high frequencies are performed more frequently, and that the nonbonding interactions, such as the van der Waals and the Coulomb interactions, are calculated less frequently. Among several multiple time-step integrator algorithms, the reversible reference system propagator algorithm (r-RESPA) proposed by Tuckermann and coworkers has been the most successfully applied.[21, 22] The versatility of the r-RESPA allows stable and precise simulations by constructing optimal integrators for each system and by adjusting the length of the time steps to integrate each motion. Martyna et al. developed the r-RESPA for various systems: (a) isothermal simulations using the Nosé–Hoover chain equations; (b) isothermal–isobaric simulations with isotropic box fluctuations using the Andersen–Hoover equations; and (c) isothermal–isobaric simulations with full-box fluctuations using the Parrinello–Rahman equations.[23, 24] The r-RESPA and extensions of the algorithm are expected to become essential components of large-scale MD simulations that have many degrees of freedom or that are run for long simulation times.

In this article, we present an efficient implementation of the canonical MD simulation by the integrated calculation of the r-RESPA with the PMEM and with the MEFMM. By analyzing the performance of the each method, we then determined the optimal implementation of each method. Extensive comparisons of the integrated methods showed useful guidelines for practical implementation of the methods. Our analysis also shows that these integrated calculations are expected to lead to significant reduction of the computational cost in complex and large MD simulations. This article is organized as follows. The next section briefly reviews the methods for the acceleration techniques. The Computational Details section discusses the computational details of our implementation. The Results and Discussion section discusses first the accuracy and efficiency of the each method for the calculation of the Coulomb interaction, and then discusses the reliability and efficiency of the integrated calculations of the r-RESPA combined with the PMEM and with the MEFMM. The Conclusion gives the key results and conclusions from the implementation of these methods.

Methods

EWALD METHOD AND PARTICLE MESH EWALD METHOD

The long-ranged Coulomb potential energy for a system with a periodic boundary is expressed in the Ewald method as a sum of the rapid convergence series in real space, V_r^{Ew}, and in reciprocal space, V_k^{Ew}, and the self-correction term, V_s^{Ew}.[1, 3, 4] The force in the Ewald method is also a sum of the contributions from the real space $\mathbf{F}_r^{\mathrm{Ew}}$ and from the reciprocal space $\mathbf{F}_k^{\mathrm{Ew}}$, each of which is easily derived by differentiating the corresponding potential energy term. The series in V_r^{Ew} and $\mathbf{F}_r^{\mathrm{Ew}}$ converges faster as α increases, which means that a small cutoff in the real space sum, $\mathbf{R}_{\mathrm{cut}}$, is sufficient to meet a given accuracy. In the Ewald method, α determines the width of the Gaussian distribution for the ith charge, q_i, such as $\rho_i(r) = q_i\alpha^3\pi^{-3/2}e^{-\alpha^2 r^2}$. In this situation, the series in V_k^{Ew} and $\mathbf{F}_k^{\mathrm{Ew}}$ converges slower, and a larger cutoff in the reciprocal space sum, M_{cut} is required to meet the given accuracy. On the other hand, the small α causes slow convergence of the real space sum and fast convergence of the reciprocal space sum. The performance of the Ewald method depends on the load balance of the two calculations, and can be controlled by the choice of α, $\mathbf{R}_{\mathrm{cut}}$, and $\mathbf{M}_{\mathrm{cut}}$. The determination of the optimal value of the parameters to minimize the CPU time, while maintaining a given accuracy, is important for consistent comparison of the performance of this and other methods. We describe this optimization procedure in a later section.

The real-space sum in the Ewald method can be implemented efficiently by using the optimal cell-list (OCL) technique in which the size of a cell is optimized to maximize the performance.[25] For the reciprocal space sum, several techniques have been proposed to reduce both the complexity and the computational effort of the calculation.[6–8] Most techniques use FFT techniques, and among these techniques the PMEM is the most promising because of its superior accuracy and reduced computational cost. Detailed derivations of the PMEM are available elsewhere,[7, 9, 10] so that here we only provide an overview of the method. Whereas the real space sum and the self-correction term in the PMEM are calculated in the same way as in the Ewald method, the rapid evaluation of the recip-

rocal space sum using the FFT permits rapid calculation of the Coulomb interaction. For consistent comparison of the performance of this and other methods, the load balance between the real-space sum and the reciprocal-space sum must be optimized, similar to the optimization required in the Ewald method. Compared with that in the Ewald method, the relative weight of the two calculations in the PMEM is changed; the rapid evaluation of the reciprocal-space sum decreases the load balance of the calculation for the real-space sum, which shifts α to larger values. In the calculation of the reciprocal-space sum in the PMEM, the interpolation method to attribute the charges to the grid for the FFT, and the size of the grid affects both the accuracy and the computational efficiency. The Cardinal B-spline interpolation has been found to be more accurate than the Lagrangian interpolation of the same order.[9] Thus, in this work we used the B-spline interpolation in the PMEM. To determine the optimal parameters, in addition to optimizing α and $\mathbf{R}_{cut}$, we also optimized the order of the interpolation and the size of the grid.

FAST MULTIPOLE METHOD

A comprehensive description of the FMM is available elsewhere.[11–20] The FMM proposed by Greengard and Rokhlin[11] begins by dividing the simulation system, referred to as level 0, into eight small boxes by bisecting each side. The resulting system of boxes is referred to as level 1, and each box divided is successively divided into eight smaller boxes, which is then referred to as level 2. This space decomposition is repeated until each box at the highest level includes less than a given number of particles. The Coulomb potential energy is expressed as the sum of the direct interaction of each particle with other particles in the same smallest boxes and with particles in their neighboring boxes, referred to as direct sums, and of the interaction with the local field produced by the truncated multipole expansion of the distant charges located in the rest of the boxes. The effective combination of the hierarchical space decomposition of the system and the multipole expansion enables very rapid evaluation of the Coulomb interaction. The accuracy and the efficiency of the FMM is determined by the region of the direct sum and by the length of the multipole expansion representing the distant charge distributions. The longer term expansion, the smaller region of the direct sum is enough for a given accuracy and *vice versa*. For reliable comparison of the computational effort of this and other methods, these parameters must be determined for a given simulation accuracy.

The FMM originally proposed was for systems with free boundary conditions.[11, 12] The extension to systems with periodic boundary conditions has been proposed and refined by several authors.[15, 19, 20] Figueirido et al. developed the FMM using the Ewald sum technique to treat periodic systems (referred to as periodic FMM or PFMM), which was inspired by Schmidt and Lee's work,[15] and successfully used the FMM for implementing MD simulations of various systems. In the PFMM, the infinite sum used to account for the periodic images can be calculated independently of the instantaneous particle coordinates, only when the size or the shape of the system does not change. Even though the calculation for the infinite sum is complicated, the advantage of the factorization of the periodic images makes the PFMM a prospective method for MD calculations with a fixed frame. An alternative form is to use the same techniques used in the multipole expansion in the original simulation system (hereafter called unit system) to form an FMM that includes the effect of macroscopic periodic assemblies of the simulation system. We refer to this method as the macroscopic expansion of the FMM (MEFMM). The macroscopic expansion (ME), including the effect of the replicas of the unit system, is defined as level 1. The effect of the replicas of the whole system of the ME at level 1 is incorporated into the unit system in the same way, and is referred to as level 2 of the ME. The procedure can be recursively applied to the system to incorporate the effect of an increasing number of periodic images of the unit system. As the level of the ME increases, the calculated Coulomb interaction converges on the exact limit of the infinitely periodic system. Similar to how practical implementation of the Ewald method requires truncation of the real and reciprocal space sums, the ME is truncated at a finite level appropriate for a given accuracy. The CPU time required to implement the ME is small compared with the multipole expansion in the unit system. This is because the calculation of the direct sum and the construction of the multipole moments from the charges in the boxes at the highest level are not needed; however, the three transformations between the multipole moments and the coefficients of the local field expansion are only applied to the macroscopic assemblies. Compared with the FMM, the MEFMM needed from 10 to 30% more CPU time, depending on the level of the ME; the increased CPU time was less than 10% for calculation with the MEFMM up to level 2. When the periodic images

of the unit system are incorporated by the ME, the macroscopic assemblies can be classified according to the level of the ME. The clustered macroscopic assemblies can be effectively used in the r-RESPA; the different time scales can be applied to the force arising from each cluster in the ME. In the calculation with the r-RESPA combined with the FMM, the total Coulomb force is decomposed into the forces depending on the time scales or the distance in the unit system. The extension of the force decomposition to the periodic images by using the clustering of the macroscopic assemblies is favorable for the r-RESPA for large systems. The ME can also be extended to incorporate any boundary shape; the ME of the FMM for anisotropic periodic systems can be implemented in a similar manner as that for the isotropic periodic system. In this study, we used the ME technique for the FMM, because the efficiency of the r-RESPA with the MEFMM in the MD simulations has not been discussed in detail elsewhere. In a following section we discuss the optimal choice of the length of the multipole expansion, the region of the direct sum, and the level of the ME.

REVERSIBLE REFERENCE SYSTEM PROPAGATOR ALGORITHM

The r-RESPA is based on the Trotter decomposition of the Liouville operator.[26] The time evolution of the entire system is generated by using a product of $2m - 1$ evolution operators, when time scales cause the Liouville operator to be decomposed into m parts. The versatility of the r-RESPA arises from the choices for the decomposition of the time–evolution operator and for the time scale applied to evolve each motion. The motions with small periods are evolved with small time steps, whereas those with large periods are evolved with larger time steps; the time scale for the Coulomb interaction is larger than that for the rest of the interaction. For a given overall accuracy, the interaction with large period can be recalculated less frequently than the other terms, by setting the corresponding Liouville operator to outer stages in the Trotter decomposition. The r-RESPA for the canonical MD simulation using Nosé–Hoover thermostats[27, 28] is derived from the extended Liouville operator for canonical ensembles in the same way as for microcanonical ensembles. The Liouville operator representing the thermostats, and the coupling with the thermostats iL_{NHC} is added to the Liouville operator for the N particle system. When the time evolution by $\exp[i(\Delta t_m/2)L_{\mathrm{NHC}}]$ is slower than the largest period of the motions in the N particle system, $\exp[i(\Delta t_m/2)L_{\mathrm{NHC}}]$ is operated at the beginning and at the end of the motions of the N particles. Other factorizations are also possible depending on the system, and the details are given by Martyna et al.[24]

Computational Details

We evaluated the performance of these methods for water systems; the number of particles in the systems was 3000, 9999, 30,000, 60,000, and 99,840. The sizes of the systems were determined by setting the density to 1.0 g/cm^3, for which the respective sizes were 31.03, 46.36, 66.86, 84.24, and 99.88 Å. We adopted the flexible water model based on the TIP3P model,[29, 30] in which the bond-stretching and angle-bending motions were included in the MD simulations. Because the nature of the water system was not discussed, the choice of the water model used was not important. We do not expect the use of other water models, such as the flexible SPC water model,[31] to affect the results we obtained from our MD simulations. The van der Waals radius and the homoatomic well depth for hydrogen atoms in the model were set to 0, so that only the van der Waals interaction between the oxygen atoms was evaluated. As described in the Results section, the relative errors in the Coulomb forces evaluated by the Ewald method, the PMEM, and the MEFMM were set to less than 1.0×10^{-4}. The same criterion was used for the calculation of the van der Waals force, which resulted in a cutoff of 8.0 Å. Smoothing functions[2] were used at the cutoff to taper the potential energy and the force to 0, and the OCL technique was used to accelerate the calculation.

In the calculation for the time evolution, we solved the Nosé–Hoover equations by using the successive substitution technique (SST), the velocity–verlet algorithm (VVA), and the r-RESPA. The SST is the simplest technique for solving nonlinear equations, and the VVA is derived from the r-RESPA that uses a uniform time step. In the calculation of the r-RESPA, we used the bond-stretching and the angle-bending motions of water molecules as the reference system. When the entire motions were divided into two parts, the reference system and the rest of the system were evolved according to $\mathbf{F}_1$ and $\mathbf{F}_2$, respectively, where $\mathbf{F}_1$ and $\mathbf{F}_2$ are defined as

$$\mathbf{F}_1 = \mathbf{F}_{\mathrm{stretch}} + \mathbf{F}_{\mathrm{bend}}, \quad (1)$$

$$\mathbf{F}_2 = \mathbf{F}_{\mathrm{vdw}} + \mathbf{F}_{\mathrm{Coulomb}}. \quad (2)$$

We refer to the r-RESPA that uses two forces as r-RESPA2. Several choices were available for the fur-

ther division of the van der Waals force and the Coulomb force, depending on the time scale. We refer to the r-RESPA that uses three forces, F_1 and two forces that arise from the division of F_2, as r-RESPA3. The appropriate division of the forces depends on the method used to calculate the forces. Later we present details of the separation of the forces with error estimates for each method.

In the MD simulation, the initial velocities and accelerations strongly effect the errors of the methods used in the numerical integration and in the force evaluation. The initial location and configuration of the water molecules in the series of the systems were randomly generated, and the energy minimization using the steepest-descent algorithm and the conjugate gradient algorithm were used. Then, the systems were equilibrated for 10 ps at 300 K using the VVA with the PMEM. After the 10-ps simulation, each system was further equilibrated by each integrated method applied to evaluate the accuracy and the efficiency. The positions, velocities, and accelerations after a 1-ps canonical MD simulation at 300 K by the VVA with the Ewald method (VVA/EM) were used as the initial conditions for the evaluation of the error and the efficiency by the VVA/EM. Similarly, we calculated the initial positions, velocities, and two accelerations for the simulation by the r-RESPA2 combined with the PMEM (r-RESPA2/PMEM) and with the MEFMM (r-RESPA2/MEFMM). For the r-RESPA3/PMEM and the r-RESPA3/MEFMM, the same procedure was used to calculate the initial positions, velocities, and three accelerations. In the implementation of the Nosé–Hoover method, instead of the thermostat chain, the single thermostat was sufficient to produce the canonical distributions for the systems. The mass of the motion for the thermostat Q was set to 9×10^{-25} kcal/mol·s^2 for $N = 3000$ particles; 3×10^{-24} for 9999; 9×10^{-24} for 30,000; 1.8×10^{-23} for 60,000; and 3×10^{-23} for 99,840. These conditions resulted in motion of the thermostat on the order of 10 fs. The initial values of the additional degrees of freedom, s, and the conjugate momentum by which the thermostat couples with the N-particle system were set to 0 at the beginning of the first equilibration calculation.

As mentioned in the previous section, for reliable comparison of the efficiency of each method, optimal parameters for a given simulation accuracy must be used. In the Ewald method, although the total energy and the force should not be affected by the choice of the parameters α, R_{cut}, and M_{cut}, the performance of the calculation was strongly affected by this choice. To find the optimal set of the parameters to minimize the computational effort, we evaluated the performance of the OCL technique, as well as the conventional naive double loop (NDL) technique. In general, we expect the OCL technique to decrease the computational effort because the number of pairs of particles in the direct calculation is reduced by using the cell list. However, due to the overhead of the OCL calculation, for systems with $N < 30{,}000$, using the OCL might cause the computational expense to increase compared with the NDL. For a given accuracy in the PMEM, we optimized α, R_{cut}, the grid size for the FFT, and the order of the interpolation to minimize the CPU time. The OCL and NDL were also evaluated for calculation of the real space sum in the PMEM. The details of the optimization are summarized in the error estimate of the force calculation in a later section. For the MEFMM, four factors affect the efficiency and accuracy of the calculation: (a) the depth of the box decomposition of the unit system, which is determined by the number of particles in the box at the highest level; (b) the length of the multipole expansion to represent charge distributions in each box; (c) the length of the ME required to incorporate the effect of the number of images of the unit system (from finite to near infinite number of images); and (d) the well-separated criterion that determines the region for the direct sum. For a given accuracy, these parameters were optimized to give the best performance of the MEFMM. In another section we discuss the relation between the error and these parameters in the force calculation of the MEFMM. Finally, we discuss the accuracy of the three methods by comparing the forces, not by comparing the energies, because a relatively unknown, method-dependent constant term is included in the energy calculation, and this makes it difficult to make meaningful comparisons. The CPU time is reported for simulations on IBM RS/6000-SP Model 9076-307.

Results and Discussion

COMPUTATIONAL EFFICIENCY AND ERROR ESTIMATION OF THE EWALD METHOD

In the Ewald method, the Coulomb force is the sum of the two series, $\mathbf{F}_r^{\text{Ew}}$ and $\mathbf{F}_k^{\text{Ew}}$ as described in the previous two sections. There are several expressions for estimating the force error in the Ewald method for a given set of α, $\mathbf{R}_{\text{cut}}$, and $\mathbf{M}_{\text{cut}}$. The total relative error, Δf^{Ew}, is the sum of the errors in the real space sum, ΔF_r^{Ew}, and in the reciprocal space

sum, ΔF_k^{Ew}, and can be evaluated as

$$\Delta f^{\mathrm{Ew}} = \frac{\Delta F_r^{\mathrm{Ew}} + \Delta F_k^{\mathrm{Ew}}}{\sqrt{\frac{1}{N}\sum_{i=1}^{N} |\mathbf{F}_i|^2}}, \tag{3}$$

where $\mathbf{F}_i$ is the Coulomb force exerted on ith particle. We calculated ΔF_r^{Ew} and ΔF_k^{Ew} as

$$\Delta F_r^{\mathrm{Ew}} \cong \frac{2\sum_i q_i^2}{\sqrt{NR_{\mathrm{cut}}L^3}} \exp\left(-\alpha^2 R_{\mathrm{cut}}^2\right), \tag{4}$$

$$\Delta F_k^{\mathrm{Ew}} \cong \frac{2\alpha\sum_i q_i^2}{L}\left(\frac{1}{\pi M_{\mathrm{cut}} LN}\right)^{1/2} \times \exp\left(-\frac{\pi^2 M_{\mathrm{cut}}^2}{\alpha^2}\right), \tag{5}$$

where L was the side length of the system.[32]

We evaluated the optimal parameters for each system as follows: (1) R_{cut} was evaluated, starting at $L/2$ and decreasing the value in steps of 0.5 Å. (2) The optimum α was calculated for each R_{cut} subject to the constraint that eq. (4) was within a given error criterion. (3) The optimal M_{cut} for each set of α and R_{cut} was calculated subject to the constraint that eq. (5) was within a given error criterion. (4) For these values of α, R_{cut}, and M_{cut}, the total CPU time was measured.

The criterion used in eqs. (4) and (5) was set to 1.0×10^{-3} kcal/mol, which resulted in a total error less than 1.0×10^{-4} in eq. (3). The criterion for the total force error was also applied in the error estimation of the PMEM method and of the MEFMM. The optimal parameters used to minimize the computational effort of the MD calculations are listed in Table I.

The computational effort was strongly affected by the choice of the parameters, and increased as N in the system increased. Although the OCL resulted in decreased computational effort for $N > 30{,}000$, due to the overhead required to make a list of particles in the system for the OCL, for $N < 30{,}000$ the NDL was superior to the OCL.

COMPUTATIONAL EFFICIENCY AND ERROR ESTIMATION OF THE PMEM

Similar to the total force error in the Ewald method, that in the PMEM was estimated by using eq. (3). Equation (4) also uses estimation for the force error in the real space sum, because $\mathbf{F}_r^{\mathrm{Ew}} = \mathbf{F}_r^{\mathrm{PME}}$. We calculated the force error in the reciprocal space sum as

$$\Delta F_k^{\mathrm{PMEM}} = \sqrt{\frac{\sum_i^N |\mathbf{F}_i^{\mathrm{recip}} - \mathbf{F}_{ki}^{\mathrm{PMEM}}|^2}{N}}, \tag{6}$$

where $\mathbf{F}_i^{\mathrm{recip}}$ is the exact reciprocal force exerted on the ith particle, within machine accuracy, and $\mathbf{F}_{ki}^{\mathrm{PMEM}}$ is the reciprocal force calculated by the PMEM. The $\mathbf{F}_i^{\mathrm{recip}}$ was calculated in the same way as that in the Ewald method with large M_{cut}, which, for a given α, was determined from eq. (5) such that $\Delta F_k^{\mathrm{Ew}} < 1 \times 10^{-15}$. The orders of the B-spline we examined were 12, 10, 8, 6, and 4. To achieve optimum performance of the FFT, the grid size should be a combination of powers of 2, 3, or 5; the range we investigated was from 16 ($=2^4$) to 200 ($=2^3 \times 5^2$). We determined the optimal set of the parameters for each system as follows: (1) R_{cut} was evaluated, starting at $L/2$ and decreasing in value in steps of 0.5 Å. (2) The optimum α was calculated for each R_{cut} subject to the constraint that eq. (4) was within a given error criterion. (3) For each α obtained in step (2), M_{cut} was calculated from eq. (5), such that $\Delta F_k^{\mathrm{Ew}} \leq 1 \times 10^{-15}$. (4) $\mathbf{F}_i^{\mathrm{recip}}$ was calculated by the Ewald method by using α and M_{cut} obtained in steps (2) and (3). (5) For each α and R_{cut}, and for a given order of the B-spline, the optimal grid size of the FFT to minimize the total CPU time was determined, subject to the constraint that the total force

TABLE I.
Optimal Parameters for Minimizing the CPU Time, for Fixed Accuracy, for the Calculation of the Coulomb Interaction in Water Systems by Using the Ewald Method.

Number of Particles	3000	9999	30,000	60,000	99,840
α (Å^{-1})	0.22453	0.15185	0.16416	0.14124	0.12139
R_{cut} (Å)	14.0	20.5	19.0	22.0	25.5
$K_{\mathrm{cut}}L/2\pi$ [a]	7	7	11	12	11
Algorithm[b]	NDL	NDL	OCL	OCL	OCL

[a] L is the side length of each system.
[b] NDL and OCL stand for the naive double loops and the optimum cell list, respectively. See the text for details.

TABLE II.
Optimal Parameters for Minimizing the CPU Time, for Fixed Accuracy, for the Calculation of the Coulomb Interactions in Water Systems by Using the PMEM.

Number of Particles	3000	9999	30,000	60,000	99,840
α (Å^{-1})	0.64484	0.64485	0.53502	0.45687	0.49288
R_{cut} (Å^{-1})	5.00	5.00	6.00	7.00	6.50
Order of B-spline	8	8	6	6	6
Grid size of FFT	48	72	100	108	135

error from eq. (3) was less than a predetermined error. (6) For each α, R_{cut}, and grid size, the optimal order of the B-spline to minimize the total CPU time was determined subject to the constraint that the total force error from eq. (3) was less then a predetermined error. The optimal total CPU time for each set of optimal α, R_{cut}, grid size, and order of the B-spline were tabulated. (7) The set of parameters that minimized the total CPU time was determined from the list made in step (6).

In Table II, the optimized values of α, R_{cut}, the grid size, and the order of the B-spline are tabulated for the series of the systems.

Compared to the calculation with the Ewald method, the optimal R_{cut} was very small for all the systems. Compared with the contribution of $\mathbf{F}_r^{\mathrm{Ew}}$ to $\mathbf{F}^{\mathrm{Ew}}$, the relative weight of $\mathbf{F}_r^{\mathrm{PME}}$ to $\mathbf{F}^{\mathrm{PME}}$ decreased while the weight of $\mathbf{F}_k^{\mathrm{PME}}$ to $\mathbf{F}^{\mathrm{PME}}$ increased. This is because the PMEM calculated the reciprocal contribution more efficiently than did the Ewald method. For systems composed of more than 10,000 particles, even the most accurate calculation in our implementation, where the orders of the B-spline and the grid size of the FFT were set to 12 and 200, respectively, did not yield forces accurate enough to meet the error criterion when $\alpha > 0.6$ (depending on the system). Therefore, in the application of the PMEM to large-scale calculations, the proper choice of the parameters is important to maintain the accuracy of the calculations.

COMPUTATIONAL EFFICIENCY AND ERROR ESTIMATION OF THE MEFMM

For efficient implementation of the FMM, it is desirable that a few particles are contained in each box at the highest level. However, this condition is not always met when the oct-tree spatial decomposition is used. In our implementation of the MEFMM, systems with 3000 and 9999 particles were decomposed up to level 4. The decomposition up to level 5 was used for systems with 30,000 and 60,000 particles. For the system with 99,840 particles, the decomposition was extended up to level 6. The average number of particles in the box at the highest level was 5.9 for the systems with 3000 particles; 19.2 for 9999; 7.3 for 30,000; 14.6 for 60,000; and 3.0 for 99,840. The distribution of the charges to each box was not determined by the particle coordinate but by the center of mass of the molecule, thus guaranteeing that the neutrality of each divided box was maintained. When we use the center of mass of the molecule, the size of the molecule should be small compared with the smallest box size. The water model met this requirement. This choice generated stable simulations, as is demonstrated in the next section.

Although several formulae have been proposed to estimate the maximum force error in the PFMM,[33, 34] there has not been a reliable formula to estimate the error in the MEFMM. In our implementation of the MEFMM, the force errors were estimated by comparing the results from the MEFMM with those by large-scale direct calculation. The relative force error was calculated as

$$\Delta f^{\mathrm{MEFMM}} = \sqrt{\frac{1}{N}\sum_i \left|\mathbf{F}_i^{\mathrm{MEFMM}} - \mathbf{F}_i^{\mathrm{Direct}}\right|^2} \Bigg/ \sqrt{\frac{1}{N}\sum_i \left|\mathbf{F}_i^{\mathrm{Direct}}\right|^2}, \quad (7)$$

where $\mathbf{F}_i^{\mathrm{MEFMM}}$ and $\mathbf{F}_i^{\mathrm{Direct}}$ are the force exerted on ith particle as calculated by the MEFMM and by the direct method, respectively. In the direct method of eq. (7), the number of particles in the whole system sharply increased with increasing region of the ME. For a system with 10,000 particles, the mirror image of the particles produced by the ME caused the total number of particles to easily exceed 10^8 particles. Thus, the estimation using eq. (7) can be applied to systems with up to 10,000 particles in practice. The entire system for the direct calculation that was comparable to the MEFMM with 9999 particles, was constructed by assembling $21 \times 21 \times 21$ replicas of the unit system. This yielded a cube with sides of

975 Å in length. The region of the ME described below was thus covered by the entire system of the assembled replicas. On the order of 30 days of CPU time was required even for this calculation. On the other hand, for a system with 9999 particles, the average number of particles in the box at the highest level was the largest among those in the systems we tested. The Coulomb potential surface could be assumed to be more complex if a larger number of molecules is included in the box. Therefore, the parameters determined to meet the force error criterion could be reasonably applied to the systems with fewer particles in the box at the highest level. The optimal parameters determined for the system with 9999 particles were used for the calculations with the MEFMM.

There have been several definitions for the well-separated criterion that determines a group of boxes included in the direct sum in the FMM.[11–19, 33–37] According to the definition proposed by Warren and Salmon, the well-separated condition was defined by using the distance between the smallest spheres including in the boxes.[35] Given a box A at the highest level, box B was defined as well-separated if $r_A + r_B \geq \eta \cdot R_{AB}$ was satisfied. Here r_A and r_B were the radii of the smallest spheres including in boxes A and B, respectively, R_{AB} was the distance between the center of the two spheres, and η was a number such that $0 < \eta < 1$. In our search for the optimal parameters $\eta < 0.5$ was disregarded because it caused the region for calculating the direct sum to include the third and fourth nearest-neighbor boxes, which in turn, caused unacceptably large computational effort. The value of η also affected the size of the region for the ME, because the region was determined to have same shape as the region for the direct sum, which enabled efficient calculation of the ME of the FMM. The region of the ME at level 2 was extended to the 10th nearest neighbor replicas of the unit system, the size of which was large enough to converge the Coulomb interaction. Thus, the ME was examined up to level 2.

In Figure 1, the CPU time to implement the MEFMM is shown as a function of the relative force error calculated with eq. (7).

Only the results with the ME up to level 2 are shown in Figure 1, because the ME up to level 1 was slightly less efficient than the ME up to level 2 when the force was calculated to a given accuracy; the same lines slightly shifted toward the upper right were obtained from the results of the ME up to level 1. In the calculation using the multipole expansion up to a quadrupole, the errors were larger than the acceptable criterion of 1×10^{-4} for

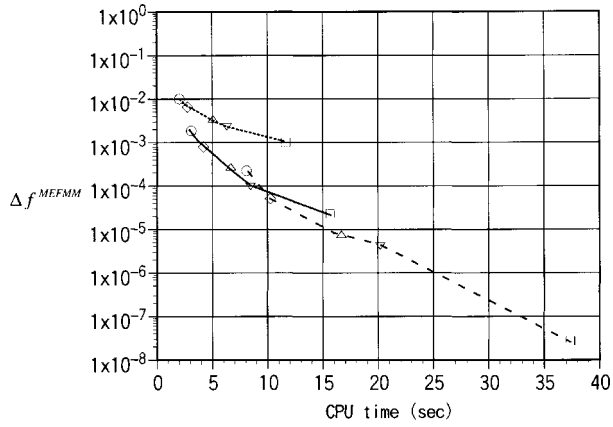

FIGURE 1. Relative force error and CPU time required for calculating the Coulomb interaction by using the MEFMM with macroscopic expansion up to level 2 for a system with 9999 particles. Results involve multipole expansion up to a quadrupole (solid line), an octupole (broken), and a hexadecapole (dotted). Calculations are for $\eta = 0.9$ (○), 0.8 (◇), 0.7 (△), 0.6 (▽), and 0.5 (□).

all the well-separated criterion examined. To meet this criterion, the multipole expansion up to an octupole was faster than that up to a hexadecapole. When we required the force error less than 1×10^{-5}, only the calculation with the multipole expansion up to a hexadecapole was found to meet the criterion. In our comparison of the efficiency of the methods to calculate the Coulomb interaction, the criterion of the relative force error was set to less than 1×10^{-4}. Therefore, the calculation by the multipole expansion up to an octupole was selected. For these conditions, the optimal $\eta = 0.6$. The size of the entire system with ME at level 2 and $\eta = 0.6$ extended over 700 Å, and the effect up to the seventh nearest neighbor of the replicas of the unit system was included. In the calculation of the MD simulation by the r-RESPA/MEFMM, the optimal parameters obtained above were used.

RELIABILITY OF THE r-RESPA COMBINED WITH THE PMEM AND WITH THE MEFMM IN THE CANONICAL MD SIMULATION

The Hamiltonian, H, of the system represented in terms of the virtual variables should be conserved in the canonical MD simulation using the Nosé–Hoover thermostat. The conservation of the transformed quantity, H', defined as

$$H' = \sum_i \frac{\mathbf{p}_i'^2}{2m_i} + V(\mathbf{q}') + \frac{s'^2 p_s'^2}{2Q} + g k_B T \ln s, \quad (8)$$

should be required in the simulation implemented by using the equations expressed in terms of the real variables, where $\mathbf{p}'_i$ is the conjugate momentum of the ith particle, $V(\mathbf{q}')$ is the potential energy, s represents the coordinate to couple with the thermostat, p'_s is the conjugate momentum of s, Q is a parameter that behaves as a mass for the motion s, k_B is the Boltzmann constant, T is a given temperature for the system, and g must be set to $3N$ for this case. The primes indicate that the variables are in terms of the real variables. Similar to how the conservation of the total energy during the MD calculation indicates the reliability of the simulation in the microcanonical ensemble, the reliability in the canonical MD simulation is evaluated by using the accumulated relative deviation of H', defined as

$$\Delta H' = \frac{1}{N_{\text{step}}} \sum_{i=1}^{N_{\text{step}}} \left| \frac{H'_i - H'_1}{H'_1} \right|. \qquad (9)$$

For systems with 9999 particles, $\Delta H'$ calculated for a 1 ps canonical MD simulation is shown in Table III.

The quantity $\log \Delta E \leq -2.5$ indicates the acceptable numerical accuracy for the time evolution in constant-energy MD simulations, where ΔE is the accumulated relative deviation of the total energy as defined in eq. (9).[38] This criterion can be reasonably applied to the canonical MD simulation by substituting ΔE with $\Delta H'$. Because the efficiency of the combination of r-RESPA with the Ewald method (r-RESPA/EM) is described elsewhere,[39] and because the r-RESPA/PMEM and the r-RESPA/MEFMM are expected to yield higher performance, we did not evaluate the efficiency of the r-RESPA/EM. In the calculation using the PMEM, the optimal R_{cut} was sufficiently small that a single time step was used to calculate the evolution of the force arising from the real space sum. For the force arising from the reciprocal space sum, a single time step was used in the time evolution of the force, because the division of the force based on the time scales was difficult. The time scale of the force from the reciprocal space sum was found to be slower than the time scale of the force arising from the real space sum; therefore, the former force was used in the outer stage in the r-RESPA.[22, 38] In our implementation of the r-RESPA3/PMEM, the forces arising from the real space sum and the reciprocal space sum were applied in the middle stage and in the outer stage, respectively. In the

TABLE III.
For Canonical MD Calculations of Systems with 9999 Particles, the Accumulated Relative Deviation of $\Delta H'$, and the CPU Time Required for 1-ps Simulations.

	Ewald		PMEM		MEFMM	
Time Scales for Integration (fs)[a]	$\log \Delta H'$	CPU Time ($\times$100 s)[c]	$\log \Delta H'$	CPU Time ($\times$100 s)[c,d]	$\log \Delta H'$	CPU Time ($\times$100 s)[c,d]
0.25(S)[b]	−2.89	3096	−3.46	231	−4.51	399
0.25(V)[b]	−3.21	3090	−3.25	235	−4.98	401
0.25/0.50			−3.68	116	−4.96	197
0.25/1.00			−4.01	59	−4.98	99
0.25/2.00			−4.08	30	−4.54	50
0.25/4.00			−3.42	15	−4.10	27
0.25/8.00			∞[e]	—	∞	—
0.25/1.00/2.00			−3.96	44	−5.03	83
0.25/1.00/4.00			−3.89	37	−5.00	73
0.25/1.00/8.00			∞	—	−2.89	69
0.25/2.00/4.00			−3.52	22	−4.52	42
0.25/2.00/8.00			∞	—	−2.61	37
0.25/4.00/8.00			∞	—	−2.04	21

[a] Time scales used in the innermost stage (the reference system), middle stage, and outermost stage are separated by forward slashes.
[b] (S) and (V) indicate that successive substitution and the velocity-verlet algorithm were used to solve the equations, respectively. The unmarked rows are the results by using the r-RESPA.
[c] CPU time on an IBM RS/6000-SP Model 9076-307.
[d] Averages over five measurements.
[e] ∞ indicates that the calculation failed due to a large, fatal deviation of $\Delta H'$.

r-RESPA3/MEFMM, the inner stage of the Coulomb force was derived from the direct sum and the outer stage was derived from the local field produced by the multipole expansion of the remote charges. Although further division of the outer stage, which depends on the distance, might accelerate the MD simulation for systems larger than those examined in this study, we only examined the r-RESPA2 and the r-RESPA3.

In Table III, the r-RESPA2(0.25/8.00) generated unacceptably large errors both for the combination with the PMEM and with the MEFMM, where r-RESPA2(X/Y) represented the r-RESPA2 with X fs and Y fs time steps for the inner and outer stages, respectively, and similar notation was also adopted for r-RESPA3. These results indicate that the time step to integrate the motion of particles in the inner stage governed by the Coulomb and the van der Waals forces should be less than 8.00 fs. For r-RESPA3(0.25/2.00/8.00) and r-RESPA3(0.25/4.00/8.00), the calculation with PMEM caused an unacceptably large error that was fatal,[40] whereas the MEFMM did not produce fatal errors; they were large compared with the errors from the other calculations. The difference in errors among these calculations was caused by the division of the Coulomb force according to the method used. In the implementation of the r-RESPA3/PMEM, although the force arising from the reciprocal space sum was treated in the outer stage, the effect from the near region was included in it, to which time steps no larger than 4.00 fs should be used. On the other hand, as the force derived from the MEFMM was divided only on the basis of the distance, a smaller time step was applied to the force from the near region and a larger time step was applied to that from the remote region. Compared to the error by using the PMEM, the error by using the MEFMM was smaller when the same r-RESPA was used. This was because the breakdown of the complete neutrality of the system caused by the cutoffs in the real space sum and in the reciprocal space sum in the PMEM amplified the fluctuation of the total energy, and because the MEFMM had no such breakdown in our simulated systems. All the cases with finite errors in Table III except for the r-RESPA3/MEFMM(0.25/4.00/8.00) provided acceptable simulations.

In the implementation of the MD simulation with the FMM, the neutrality of each divided box was found to be especially important for reducing the accumulation of the error. The level of approximation to represent the charge distribution in the FMM suddenly changed between the region for the direct sum and the rest of the region. The charges in the former were added pairwise, whereas those in the latter were evaluated by using the multipole expansion. When a given charge traverses the regions in the MD simulation, the effect of the charge on the total energy varies and an artificial charge fluctuation is produced. The fluctuation also occurs between the boxes at different levels. The difference in the approximation to represent the charge distribution causes the fluctuation and upsets the neutrality of each box. The breakdown of the neutrality produces an error, such as the deviation of H'. For water molecules that have large partial charges, the errors might become larger and cause divergence. Therefore, as mentioned in the previous section, to guarantee the neutrality of each box, the distribution of charges to each box was determined by the center of mass of the molecule instead of by the particle coordinate. Maintaining neutrality is one method for preventing artificial charge fluctuations, and other choices are possible. Similar to how the smoothing technique has been used to avoid sudden cutoff of the potential energy and of the force in conventional MD calculations, and successively produces stable MD simulations, the same technique is also effective in calculations of the FMM.[19] In Figure 2, log $\Delta H'$ is shown as a func-

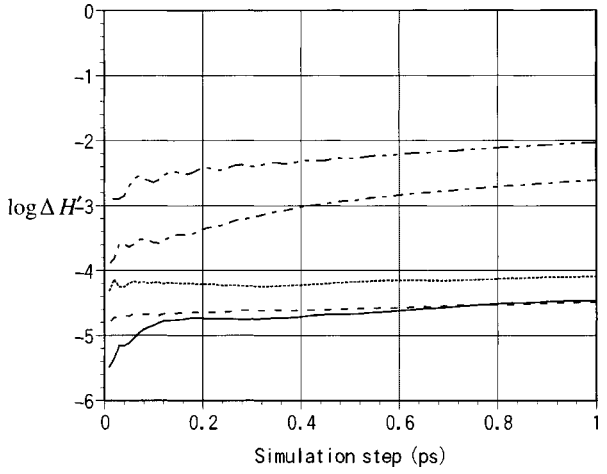

FIGURE 2. Accumulation of the relative deviation $\Delta H'$ as a function of simulation time step for a system with 9999 particles and simulated with a canonical MD simulation and using the MEFMM for the Coulomb interaction. The solid, broken, dotted, dotted broken, and double dotted broken lines represent the results by using the VVA/EM(0.25), r-RESPA2/MEFMM(0.25/4.00), r-RESPA3/MEFMM(0.25/2.00/4.00), r-RESPAS/MEFMM (0.25/2.00/8.00), and r-RESPA3/MEFMM(0.25/4.00/8.00), respectively. Acronyms are defined in the text.

tion of simulation steps in the 1-ps canonical MD simulation by the r-RESPA2/MEFMM and by the r-RESPA3/MEFMM.

The r-RESPA3/MEFMM(0.25/2.00/8.00) and the r-RESPA3/MEFMM(0.25/4.00/8.00) produced a slight drift of $\log \Delta H'$. However, the former did not exceed the error criterion, whereas the latter did. The other algorithms shown in Ftgure 2 did not produce drift and maintained the relative deviations of $\log \Delta H' < -4$ during 1-ps simulations; indicating that the time evolutions were of acceptable accuracy. Therefore, stable simulations with these algorithms can also be expected for longer periods. By comparing the results between the r-RESPA3/MEFMM(0.25/2.00/4.00) and the r-RESPA3/MEFMM(0.25/2.00/8.00), and between the r-RESPA2/MEFMM(0.25/4.00) and the r-RESPA3/MEFMM(0.25/4.00/8.00), we found that large time steps in the outer stage increased the error. On the other hand, by comparing the results between the r-RESPA3/MEFMM(0.25/2.00/8.00) and the r-RESPA3/MEFMM(0.25/4.00/8.00), and between the r-RESPA2/MEFMM(0.25/4.00) and the r-RESPA3/MEFMM(0.25/2.00/4.00), we found that increasing the time step in the middle stage caused a relatively small increase in the error. The error was a sum of the contributions from the direct sum and from the local field expansion. The above two comparisons indicate that the increase in error with increasing time step in the calculation of the local field expansion was larger than that in the direct sum, and that careful choice of the time steps reduced the error even when 8.00 fs time steps were used in the outer stage.

The calculations with the PMEM instead of the MEFMM yielded similar results. The driffs presented in the calculations by using the r-RESPA3/MEFMM(0.25/2.00/8.00) and (0.25/4.00/8.00) were increased in the calculations with the PMEM. Consequently, the r-RESPA3/PMEM(0.25/2.00/8.00) and (0.25/4.00/8.00) resulted in divergence of the relative errors (see Table III).

EFFICIENCY OF THE r-RESPA COMBINED WITH THE PMEM AND WITH THE MEFMM IN THE CANONICAL MD SIMULATION

Table IV shows the CPU times needed to perform 1-ps canonical MD simulations by using the VVA/EM, r-RESPA2/PMEM, r-RESPA2/MEFMM, r-RESPA3/PMEM, and r-RESPA3/MEFMM.

Although the timing depended on the machines, the CPU time required to implement the Ewald method was five to six times longer than that in the work by Figueirido et al., whereas the CPU time for the FMM was smaller. For the discussion of the speedup compared with the calculation using the VVA/EM(0.25), the performance of the calculations by using the PMEM and by using the MEFMM in this study might be overestimated. We note that the parameters used to implement each method were determined to calculate a force sufficiently accurate to meet the criterion. Because the formulae used in the Ewald method was for the maximum force error, the smaller cutoffs in the real-space sum and in the reciprocal-space sum might maximize the efficiency of the Ewald method. Therefore, the performance regarding the Ewald method presented in Table IV could be underestimated. Because the computational cost of the Ewald method grew substantially with increasing system size, to the point that it was not practical to perform the calculations, the CPU time for the system with 99,840 particles was extrapolated from the results for the smaller systems. We extrapolated times by using both a quadratic equation and an equation of the form $y = a \times x^{3/2} + b$, which yielded 1.14×10^7 and 1.07×10^7 s, respectively. Neither the r-RESPA2/PMEM(0.25/8.00) nor r-RESPA2/MEFMM(0.25/8.00) generated stable MD simulations for the systems with 30,000, 60,000, and 99,840 particles, similar to the systems with 9999 particles (see Table III). Although the r-RESPA3/MEFMM(0.25/4.00/8.00) resulted in the accumulation of a finite error for the system with 9999 particles, the algorithm yielded a larger error and occasionally caused divergence for larger systems. When the time steps were larger than 4.00 fs for the middle stage, unacceptably large accumulated errors were generated in the systems with 30,000, 60,000, and 99,840 particles (data not shown in Table IV).

Comparing the efficiency of the r-RESPA2 with the r-RESPA3, the latter yielded superior performance when the same time steps were used in the outer stage of the r-RESPA2 and in the middle stage of the r-RESPA3. For both the PMEM and the MEFMM, however, the fastest implementation with the r-RESPA2 was more efficient than with the r-RESPA3. The r-RESPA2/PMEM(0.25/4.00) and the r-RESPA2/MEFMM(0.25/4.00) were faster than the r-RESPA3/PMEM(0.25/2.00/4.00) and the r-RESPA3/MEFMM(0.25/2.00/8.00), respectively. The r-RESPA2/PMEM(0.25/4.00) calculated the Coulomb and van der Waals forces every 16 steps, whereas the r-RESPA3/PMEM(0.25/2.00/4.00) evaluated the inner stage of the Coulomb force and the van der Waals force every eight steps, and the outer stage of the Coulomb force

TABLE IV.

CPU Time (×100 s) Required for 1-ps Canonical MD Simulation by Using the r-RESPA Combined with the PMEM or the MEFMM: The Data Calculated from a 0.2 ps MD Simulations Were Scaled.[a]

Time Scales for Integration (fs)[b]	Method for Calculating the Coulombic Interaction	Number of Particles				
		3000	9999	30,000	60,000	99,840
0.25	Ewald method[c]	463	3090	17606	49728	
0.25	PMEM[d]	62	235	681	1570	2361
0.25/0.50		32	116	340	788	1180
0.25/1.00		16	59	170	397	597
0.25/2.00		8	30	85	201	302
0.25/4.00		4	15	45	100	160
0.25/1.00/2.00		12	44	145	356	511
0.25/1.00/4.00		9	37	130	330	470
0.25/2.00/4.00		6	22	72	175	264
0.25	MEFMM[d]	160	401	1350	2291	4060
0.25/0.50		77	197	674	1147	2022
0.25/1.00		39	99	338	577	1009
0.25/2.00		19	50	171	292	503
0.25/4.00		10	27	85	153	260
0.25/1.00/2.00		25	83	227	459	872
0.25/1.00/4.00		17	73	169	397	814
0.25/2.00/4.00		12	42	116	239	440
0.25/2.00/8.00		8	37	87	205	405

[a] CPU time on an IBM RS/6000-SP Model 9076-307. The unit of the CPU time is 100 s.
[b] (S) and (V) indicate that successive substitution and the velocity-verlet algorithm were used to solve the equations, respectively. The unmarked rows are the results by using the r-RESPA.
[c] See the text for the CPU time for the system with 99,840 particles.
[d] Averages are over five measurements.

every 16 steps. As described in the previous section, the division of the Coulomb force in the PMEM caused the time step for the outer stage in the r-RESPA3/PMEM to be less than 4.00 fs, and the r-RESPA3 was the same as the r-RESPA2 when the same time step was used for both the middle and the outer stages. To improve the efficiency of the r-RESPA3/PMEM compared with the r-RESPA2/PMEM, other divisions of the Coulomb force in the PMEM are required. Compared with the calculations with the PMEM, the MEFMM enabled the r-RESPA3 to take larger time steps in the outer stage. The r-RESPA3/MEFMM(0.25/2.00/8.00) calculated the inner stage of the Coulomb force and the van der Waals force every eight steps, and the outer stage of the Coulomb force every 32 steps, whereas the r-RESPA2/MEFMM(0.25/4.00) calculated the Coulomb and van der Waals forces every 16 steps. The decrease in computational effort arising from less frequent calculation of the outer stage of the Coulomb force canceled out the more frequent calculation in the inner stage. In the calculation for larger systems, further decomposition of the outer stage in the MEFMM is possible, which may lead to more efficient implementation than that by the r-RESPA2/MEFMM(0.25/4.00). As previously mentioned, although the speedup compared with the VVA/EM(0.25) might be slightly overestimated, the performance is not expected to change significantly when the comparison is made at the optimized conditions. As the size of the system increases, the speedup became more evident. The maximum speedup of the r-RESPA2/PMEM(0.25/4.00) and r-REPSA2/MEFMM(0.25/4.00) occurred for the system with 99,840 particles: the r-RESPA2/PMEM and r-RESPA2/MEFMM accelerated the calculation by a factor of about 700 and 400, respectively. The calculations with the PMEM were faster than those with the MEFMM for systems composed of fewer than 100,000 particles. The computational effort with r-RESPA/PMEM was about half of that with the r-RESPA/MEFMM.

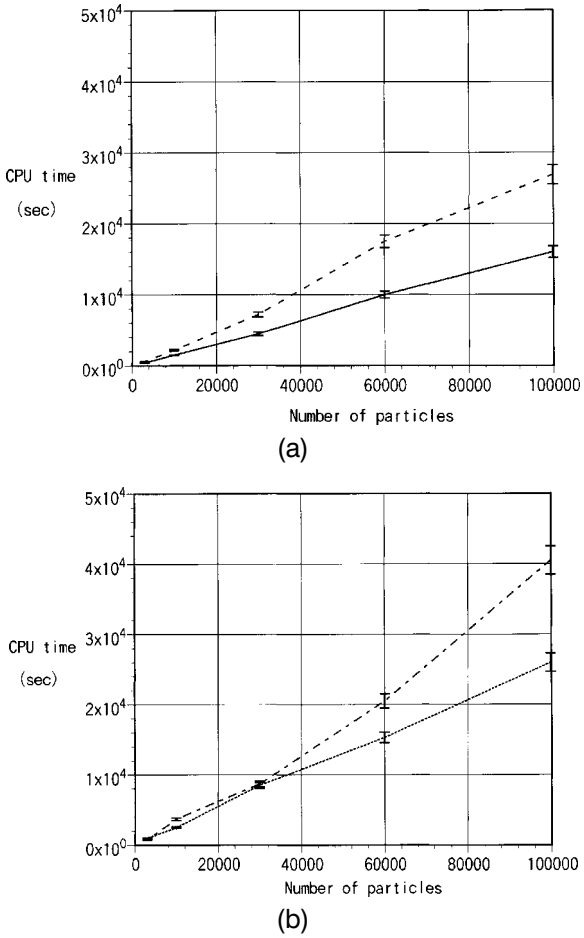

FIGURE 3. CPU time required for a 1-ps canonical MD simulation as a function of the number of particles N in the system by the r-RESPA combined with (a) the PMEM, and (b) the MEFMM. Data calculated from a 0.2-ps MD simulation is scaled and are the average value of five calculations is plotted (error bars show the variation for these five measurements). The solid, broken, dotted, and dotted broken lines represent the results by using the r-RESPA2/PMEM(0.25/4.00), r-RESPA3/PMEM(0.25/2.00/4.00), the r-RESPA2/MEFMM(0.25/4.00), and r-RESPA3/MEFMM(0.25/2.00/8.00), respectively. Acronyms are defined in the text.

The CPU time for the 1-ps canonical MD simulation with the r-RESPA2/PMEM(0.25/4.00) and r-RESPA3/PMEM(0.25/2.00/4.00), and that with the r-RESPA2/MEFMM(0.25/4.00) and r-RESPA3/MEFMM(0.25/2.00/8.00) are shown as a function of the number of particles in Figures 3a and b, respectively.

Each choice of the time steps yielded the most efficient calculation by each integrated method, while retaining stability of the MD simulation. The CPU time was measured five times and averaged to remove the error caused by the system condition of the computer used. The average values are connected by lines and the magnitudes of the error bars are less than 5% for each average value. In Figure 3a, the CPU time is scaled as $3.282\times10^{-2}\times N\log N$ ($R^2 = 0.9972$) for the r-RESPA2/PMEM(0.25/4.00), and is scaled as $5.556 \times 10^{-2} \times N\log N$ ($R^2 = 0.9935$) for the r-RESPA3/PMEM(0.25/2.00/4.00), respectively, where R^2 is the coefficient of determination in the multiple regression analysis, which indicates goodness-of-fit (1 is a perfect fit). In Figure 3b, the results for r-RESPA2/MEFMM(0.25/4.00) are scaled as $2.604 \times 10^{-1} \times N$ ($R^2 = 0.9987$), and those for r-RESPA3/MEFMM(0.25/2.00/8.00) are scaled as $3.824 \times 10^{-1} \times N$ ($R^2 = 0.9867$). Although the numerical complexity of the PMEM was proportional to $N\log N$ and that of the FMM was proportional to N, the coefficients of the scaling reflected the complexity of the calculations. The performance of the calculation by the PMEM was determined mostly by the FFT for the interpolated charges on the grid and by the direct calculation for the real-space sum. The overall performance of the calculation for the MEFMM mainly determined the efficiency of the direct calculation and of the transformation from the multipole moments to the coefficients of the local field expansion (M2L transformation). For calculations of systems with 9999 particles, the cutoff distance of the real space sum in the PMEM was 5 Å, whereas the region of the direct sum in the MEFMM extended from 11 to 17 Å. The load balances of the direct calculation indicate the numerical complexity required to implement the FFT in the PMEM and the M2L transformation in the MEFMM, because they were optimized for the best performance of the overall calculation. With increasing efficiency of the FFT in the PMEM, the load-balance of the direct calculation decreases. The increase in the efficiency of the M2L transformation also decreases the load balance of the direct calculation in the MEFMM. The difference in the complexity between the FFT and the M2L transformation affected the difference in the overall performance between the PMEM and the MEFMM. The M2L transformation requires more complex operations than the FFT; therefore, the PMEM is faster than the MEFMM. The operation count of the M2L transformation is discussed elsewhere,[35, 36] and improvement of the algorithm for the M2L transformation is required to accelerate the MEFMM. The performance of the M2L transformation also affects the relative efficiency between the MEFMM and the PFMM. For the evaluation of periodic images, the MEFMM uses the M2L trans-

formation extended to the macroscopic assemblies, whereas the PFMM uses the Ewald techniques. The latter is expected to yield superior performance for systems without changes in the shape. For constant-pressure simulations, the relative efficiency of the two methods is determined by the numerical complexity of both the M2L transformation and of the Ewald sum for the PFMM. Although the M2L transformation seems to be simpler, optimized algorithms for them are required for reliable discussion of the efficiencies, and we are currently doing a detailed comparison. In addition to the performance of each method itself, the possibility for improving the computational effort when combined with the r-RESPA is important for doing large-scale simulations. The periodic images derived in the MEFMM can be classified according to the level of the ME; thus, the force decomposition based on the ME is possible. The hierarchical structure of the time evolution used in the r-RESPA and that of the space decomposition derived in the MEFMM can be effectively integrated to accelerate the calculation.

The only systems examined in this study were homogeneous water systems of various sizes. For other applications, systems such as biological macromolecules in solutions, similar to the water system, the integrated methods are expected to yield fast and stable simulations. For the water system we examined, the Coulomb interaction significantly contributed to the total energy. The error in the calculation of the Coulomb interaction strongly influenced the accumulation of the total error in the MD simulation. The partial charges on the model of each amino acid that composes the proteins are generally smaller than those on water molecules. In applications of these systems, the contribution of the Coulomb interaction is not relatively as large as that of the homogeneous water system. Therefore, the error arising from the Coulomb interaction in such applications is expected to affect the total error less than the homogeneous water system. Under such conditions, when a desired accuracy of the simulation is required, the less accurate calculation for the Coulomb interaction is sufficient. As the desired accuracy of the calculation by the PMEM and the FMM is decreased, the computational efficiency becomes more evident. In Figure 1, the CPU time quickly decreased with increasing force error, $\Delta f^{\mathrm{MEFMM}}$. Moreover, although the number of divisions of the force used in the r-RESPA is limited to three in the water system, further division of the force is possible for protein systems. Depending on the time scales, increasing the types of interactions in the system increases the possible divisions of the force, which increases the performance of the r-RESPA and of the integrated method. In the calculation for still larger and more complicated systems than those treated in this article, parallel calculations should be used. In a comparison of the efficiency of the r-RESPA/PMEM with the r-RESPA/MEFMM using parallel computing, the superiority of the r-RESPA/PMEM is not guaranteed because of the complexity of the communication between the processes in a parallel computer. We are, therefore, currently making comparisons of the computational effort for these methods on a parallel computer.

Conclusions

We studied formulation of computationally efficient canonical MD algorithms by combining r-RESPA with PMEM and with MEFMM, and evaluated the performance of the integrated methods for water systems with the number of particles ranging from 3000 to 99,840. To make consistent comparisons, for a given simulation accuracy, the conditions for minimizing the computational effort with the Ewald method, the PMEM, and the MEFMM, was first determined. Using the optimal conditions for each method and for each system size, the efficiency and the reliability of the integrated methods were evaluated.

To determine the optimal parameters for implementing the PMEM, we found that careful choice of the parameters was important to calculate the force with sufficient accuracy that the given error criterion can be met, because for some sets of parameters the desired accuracy was not achieved. The computational effort of the PMEM was also a strong function of the set of parameters. The optimal cutoff length in the real space sum obtained for the systems was small, and the force arising from the reciprocal space sum was difficult to divide. Thus, in our implementation of the r-RESPA3/PMEM, the Coulomb force was divided into two parts, representing the forces arising from the real space sum and from the reciprocal space sum. Although a small cutoff in the real space sum enabled optimal implementation of the PMEM itself, it might reduce the possibility of the r-RESPA. For a larger cutoff for the real space sum, which reduces the efficiency of the PMEM itself, the r-RESPA with a finer division of the force might compensate for the reduction of the efficiency of the PMEM and yield reduced computational effort for the overall MD simulation.

For the calculation with the FMM, including the periodic images, there are two approaches: the periodic FMM using the Ewald sum techniques, and the MEFMM. We evaluated the efficiency of the calculation with the MEFMM. We found that the molecule-based distribution of the charges to the divided boxes in the FMM was important to generate stable MD simulations; otherwise unacceptable drift in the forces and in the potential energies occurred. The efficiency of the calculation with the MEFMM also strongly depended on the parameters used, similar to the PMEM; for a given simulation accuracy, the computational effort varied by a factor of 3 to 4. Compared with the PMEM, the MEFMM enabled the r-RESPA to take larger time steps in the calculations.

Compared with the VVA/EM(0.25), the speedup of the calculation with both the r-RESPA/PMEM and the r-RESPA/MEFMM increased as the system size increased; the maximum efficiency was obtained for the largest system examined. With the r-RESPA/PMEM, the speedup was a factor of 100 for the system with 3000 particles and increased to a factor of 700 for the system with 99,840 particles. The calculation was also about 400 times faster with the r-RESPA/MEFMM. Therefore, the r-RESPA/PMEM yielded superior performance to the r-RESPA/MEFMM by a factor of about 2. Although the CPU times for the 1-ps canonical MD simulation scaled as $N \log N$ for the r-RESPA/PMEM and as N for the r-RESPA/MEFMM, where N is the number of particles, the coefficients of the scaling were significantly different. The difference originates from the difference in the numerical complexity of the FFT in the PMEM and from the M2L transformation in the MEFMM. The latter requires more complex operations than the former. The difference in the efficiency directly affects the overall performance of the integrated method.

In the canonical MD simulation with the r-RESPA for the Nosé–Hoover equations, it was important to adjust the longest time scale against the time scale for the thermostat. For the larger systems, a more appropriate choice of the time scale for the thermostat is available, and probably significantly reduces the computational cost. For the calculation of still larger systems than those we examined, computations on parallel computers is essential. We are currently comparing the efficiency of the parallel implementation of the r-RESPA combined with the PMEM and with the MEFMM. The integrated method is suitable for large-scale MD simulations, and in the near future, sophisticated implementations will permit realistic MD calculations with millions of particles and for longer simulation times.

Acknowledgments

Some of the calculations were done on an IBM SP2 at the Tsukuba Advanced Computing Center of the Agency of Industrial Science and Technology.

References

1. Allen, M. P.; Tildesley, D. J. Computer Simulation of Liquids; Oxford: Oxford Univ. Press, 1989.
2. Steinbach, P. J.; Brooks, B. R. J Comp Chem 1994, 15, 667.
3. Toukmaji, A. Y.; Board, J. A., Jr. Comput Phys Commun 1996, 95, 73.
4. Karasawa, N.; Goddard, W. A., III J Phys Chem 1989, 93, 7320.
5. Perram, J. W.; Petersen, H. G.; De Leeuw, S. W. Mol Phys 1998, 65, 875.
6. Hockney, R. W.; Eastwood, J. W. Computer Simulation Using Particles; London: IOP Publishing Ltd., 1989.
7. Darden, T.; York, D.; Pedersen, L. J Chem Phys 1993, 98, 1008.
8. York, D.; Yang, W. J Chem Phys 1994, 101, 3298.
9. Essmann, U.; Perera, L.; Berkowitz, M. L.; Darden, T.; Lee, H.; Pedersen, L. G. J Chem Phys 1995, 103, 8577.
10. Petersen, H. G. J Chem Phys 1995, 103, 3668.
11. Greengard, L.; Rokhlin, V. I. J Comp Phys 1987, 73, 325.
12. Greengard, L.; Gropp, W. D. Math Applic 1989, 20, 63.
13. Appel, A. SIAM J Sci Stat Comp 1985, 6, 85.
14. Bernes, J.; Hut, P. Nature 1986, 324, 446.
15. Schmidt, K. E.; Lee, M. A. J Stat Phys 1991, 63, 1223.
16. Ding, H.; Karasawa, N.; Goddard, W. A., III J Chem Phys 1992, 97, 4309.
17. White, C. A.; Head–Gordon, M. J Chem Phys 1994, 101, 6593.
18. Zhou, R.; Berne, B. J. J Chem Phys 1995, 103, 9444.
19. Figueirido, F.; Levy, R. M.; Zhou, R.; Berne, B. J. J Chem Phys 1997, 106, 9835.
20. Lambert, C. G.; Darden, T. A.; Board, J. A., Jr. J Comput Phys 1996, 126, 274.
21. Tuckerman, M.; Berne, B. J.; Martyna, G. J. J Chem Phys 1992, 97, 1990.
22. Tuckerman, M.; Berne, B. J.; Martyna, G. J. J Chem Phys 1991, 94, 6811.
23. Martyna, G. J.; Klein, M. L.; Tuckerman, M. J Chem Phys 1992, 97, 2635.
24. Martyna, G. J.; Tuckerman, M.; Tobias, D. J.; Klein, M. L. Mol Phys 1996, 87, 1117.
25. This technique is the same as the highly optimized algorithm developed by Perram et al. in ref. 5, which was inspired by the linked-cell list devised by Hockney and Eastwood in ref. 6.

26. Trotter, H. F. Proc Am Math Soc 1959, 10, 545.
27. Nosé, S. J Chem Phys 1984, 81, 511.
28. Hoover, W. H. Phys Rev A 1985, 31, 1695.
29. Lorgensen, W. L.; Chandrasekhar, J.; Madura, J. D.; Impey, R. W.; Klein, M. L. J Chem Phys 1983, 79, 926.
30. Dang, L. X.; Pettitt, B. M. J Phys Chem 1987, 91, 3349.
31. Zhu, S. B.; Fillingim, T. G.; Robinson, G. W. J Phys Chem 1995, 95, 1002.
32. Kolafa, J.; Perram, J. W. Mol Simul 1992, 9, 351.
33. Sølvason, D.; Petersen, H. G. J Stat Phys 1997, 86, 391.
34. Esselink, K. Comput Phys Commun 1995, 87, 375.
35. Warren, M. S.; Salmon, J. K. In Supercomputing '93, IEEE Comp Soc 1993, 12.
36. Singh, J. P.; Holt, C.; Hennessy, J. L.; Gupta, A. In Supercomputing '93, IEEE Comp Soc 1993, 54.
37. Rankin, W. T.; Board, J. A., Jr. Tech Rept 95-002, EE Dept., Duke Univ.
38. Watanabe, M.; Karplus, M. J Phys Chem 1995, 99, 5680.
39. Procacci, P.; Marchi, M. J Chem Phys 1996, 104, 3003.
40. A slight modification of the PMEM proposed by Procacci et al. could improve the errors; Procacci, P.; Darden, T.; Marchi, M. J Chem Phys 1996, 100, 10464; Procacci, P.; Marchi, M.; Martyna, G. J. J Chem Phys 1998, 108, 8799.